ELSEVIER
Available online at www.sciencedirect.com
SCIENCE DIRECT®
Science of Computer Programming 51 (2004) 291–292
Science of Computer Programming
www.elsevier.com/locate/scico

Book review

Process algebra with timing
J.C.M. Baeten and C.A. Middelburg (Eds); Springer, Berlin, 2002, pp. XX+287, ISBN 3-540-43447-X.

Over the past decade, there have been significant advances in the area of formal methods that concerns itself with the specification and verification of timed systems. Many of the formalisms, methods, and techniques that had previously been used strictly in relation to untimed systems, have been extended to incorporate timing aspects as well. This book, somewhere in between a monograph and an advanced-level textbook, narrowly focuses on one of these methods, an algebraic one.

The method used is that of the Algebra of Communicating Processes [1–3], abbreviated ACP, and often just called "process algebra" by authors from the ACP school. Process algebra (a term I will strictly use here in the narrow sense of the ACP method) is a formalism that emphasizes the definition of and calculation with processes by means of equational logic. No fixed process model is imposed, and indeed many models have been proposed and studied. In practice however, mostly a bisimulation model based on structural operational semantics is used.

The authors start out with a general introduction to untimed process algebra, and then gradually incorporate timing aspects. In particular, two directions of research are methodically explored. Firstly, the choice of the time domain, where they examine both a discrete domain and a dense domain (basically modeling moments in time as natural numbers, and as real numbers). Secondly, the distinction between relative and absolute timing is made, the difference being whether references to time are specified with respect to an "absolute clock", or with respect to the amount of time that has passed since the last event within the process. All resulting four possibilities (discrete-relative, discrete-absolute, dense-relative, and dense-absolute) are studied in detail in four corresponding chapters, and in a very uniform manner indeed, which enables the reader to compare all four directions closely, and understand the issues involved. This is all the more encouraged by the authors, who go to great lengths to show the relations between the various theories. Finally, the authors conclude with two chapters that show how the framework set out in the previous chapters can be extended with other well-known concepts from process algebra, like the silent step or the state operator.

Throughout the book, the authors give a large number of example specifications and (partial) verifications, many of them non-trivial. Even while these examples are impressive, I must note that none of them are "real life" in the sense that our colleagues in industry would be impressed with them. I suppose such examples would not have fit the scope of the work, so it is only minor criticism, but I would love to see a

0167-6423/$ - see front matter
doi:10.1016/j.scico.2004.02.001

companion volume on the applications of timed process algebra, in a framework of comparable maturity. A more serious issue for me was the complete lack of references to or comparison with related work outside the ACP school (for example to the timed extensions of CCS and CSP). Especially where the tradeoffs between the many options possible are discussed, one becomes very curious on how the corresponding issues in related work have been handled. I was annoyed by this, and feel it is a painful omission, equally so whether one views this work as a textbook or as a monograph.

Although almost everything presented in their book has been published before, the authors deserve significant credit for bringing together the most successful concepts in timed process algebra within a coherent and uniform framework. Whereas isolated publications have sometimes suffered from subtle and unexplained differences in definitions and approaches with regard to each other, this book succeeds in painting the bigger picture. In my opinion this is a clear sign that timed process algebra has grown up to be a mature theory, existing not only in the strict domain of academic research, but also ready for classroom teaching, and hopefully, industrial application.

Looking upon the work as a whole, I am pleasantly surprised by the combination of academic depth on the one hand, and the great number of examples and exercises given on the other. This makes it very suitable as a classroom or self-study textbook, for advanced undergraduate students, or for researchers in need of an introduction to timed process algebra. These textbook qualities are augmented by a large appendix containing a mathematical background primer. It would be incorrect however to view this work as a textbook only; it is just as much a monograph of timed process algebra, giving a comprehensive and accessible overview of the field.

Jan Joris Vereijken
Amsterdam, The Netherlands
E-mail address: janjoris@acm.org

References

[1] J.C.M. Baeten, C. Verhoef, Concrete process algebra, in: S. Abramsky, D.M. Gabbay, T.S.E. Maibaum (Eds.), Handbook of Logic in Computer Science, Vol. IV, Oxford University Press, Oxford, 1995, pp. 149–268.

[2] J.C.M. Baeten, W.P. Weijland, Process algebra, Cambridge Tracts in Theoretical Computer Science, Vol. 18, Cambridge University Press, Cambridge, 1990.

[3] J.A. Bergstra, J.W. Klop, Process algebra for synchronous communication, Inform. Control 60 (1–3) (1984) 109–137.